



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

09/370,724

08/07/1999

BABAK NADER

CIS0019US

8423

33031 7590 04/16/2007
CAMPBELL STEPHENSON ASCOLESE, LLP
4807 SPICEWOOD SPRINGS RD.
BLDG. 4, SUITE 201
AUSTIN, TX 78759

EXAMINER

ELALLAM, AHMED

ART UNIT

PAPER NUMBER

2616

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
--	-----------	---------------

3 MONTHS

04/16/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary

Application No.

09/370,724

Applicant(s)

NADER ET AL.

Examiner

AHMED ELALLAM

Art Unit

2616

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 16 January 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-21 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-21 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____.

DETAILED ACTION

This communication is responsive to Amendment filed on 01/06/2007. The Amendment has been entered.

Claims 1-21 are pending.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

1. Claims 1-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liese et al, US (5,854,889) in view of Mann, US (6,618,854). Hereinafter referred to respectively as Liese and Mann.

Regarding claim 1, with reference to figures 1 and 2, Liese discloses:

- a network under test 20;
- custom servers (ISDN custom server 22, CG custom server, ...) that execute test cases, see column 3, lines 9-28, (Claimed at least one probe network device coupled to the network under test, the at least one probe network device configured to host at least one task type);
- Execution server 16 coupled to the custom servers (claimed an NVT server coupled to the at least one probe network device); wherein

- a user at the network under test communicates to a client machine which test or test cases are to be executed by the custom servers, the test or test cases can be edited before transmission to the execution server 16 which coordinates the execution of test cases by the custom servers, see column 3, lines 9-47. Liese further discloses that the client machine 32 includes a GUI (Graphical User Interface) that provides an interface for managing test cases (e.g. create, change delete, store access...), see column 3, lines 29-47. Liese also discloses that the client machine may access and drive one or more custom servers deployed on a network via an execution server to perform any test capable of being performed on the network. See column 5, lines 55-59. The execution server ensures the user logged into the client machine can access and drive a number of custom servers, column 7, lines 6-9. Liese further discloses supplying the edited test case to the execution server, see column 8, lines 14-18.

The difference between Liese and claim 1 is that Liese doesn't specify the execution server (claimed NVT server) is configured to translate parameters entered by a user to instruction executable by at least one customer server, and to transmit the instructions to at least one of custom servers to execute the at least one of the test cases).

However, with reference to figures 4 and 5, Mann discloses a client device connected to a WEB server 406, the web server (also called host computer) configured for translating a source code user program into object code (claimed executable

instructions) for execution by the target processor, the target processor hosting a test, see column 7, lines 10-61 and column 8, lines 31-44.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to implement the translation of parameters by the execution server into executable instructions to be executed by respective custom servers in the system of Liese as taught by Mann. The system of Liese can be modified by configuring the execution server to translate the parameters entered by Liese into executable instructions for transmission to a specific custom server for the respective test case for execution. The advantage would be the ability to provide the client of Liese with interface programs written in platform independent language, making the clients of Liese more versatile in testing the network. (See Mann, column 8, lines 11-14 and column 9, lines 1-10).

Regarding claim 2, with reference to figure 2, Liese shows a client 32 coupled to the execution server 22, wherein client 32 includes a GUI (Graphical User Interface) that provides an interface for managing test cases (e.g. create, change delete, store access...), see column 3, lines 29-47. Liese further discloses supplying the edited test case to the execution server, see column 8, lines 14-18. (Claimed an NVT client coupled to the NVT server, wherein the NVT client is configured to provide the template to the user for entering the parameters, and the NVT client is configured to transmit the parameters to the NVT server)

Regarding claim 3, with reference to figure 2, Liese shows that the execution server is coupled through the bus 12 (claimed Ethernet control network) to the custom servers (claimed at least one probe network device).

Liese doesn't disclose a communication server between the bus (Ethernet control network) and the custom servers (claimed at least one probe network device).

However, Applicants disclose that the communication server couples probe network devices to the control network 12., and that the NVT server is coupled to control network and communicates through communication server to network probes. See specification page 8, lines 11-15. (Examiner interpreted the function of such arrangement (communication server and the NVT server) as being the same function of Liese Execution server, since the execution server couples the bus and the custom servers).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to make the direct connection of Liese in view of Mann' execution server to the custom servers through another server (communication server) so that scalability to the number of customer servers to a corresponding execution server can be provided if a need arises.

Regarding claim 4, Liese discloses that a custom server is CG server (call generator), see column 3, lines 17-19, column 6, lines 34-35. (Claimed one task type includes a traffic generator).

Regarding claim 9, with reference to figures 1 and 2, Liese discloses a method for testing a network, comprising:

- providing a test network 20 having custom servers (ISDN custom server 22, CG custom server, ...) that execute test cases, see column 3, lines 9-28, (Claimed providing a test network comprising a probe network device hosting a task type and further coupled to the probe network device);
- providing an Execution server 16 coupled to one of the custom servers (probe network device) (claimed providing a NVT server coupled to the probe network device);
- a user in the network under test communicates to a client machine which test or test cases are to be executed by the custom server(s), the test or test cases can be edited before transmission to the execution server 16 which coordinates the execution of test cases by the custom servers, see column 3, lines 9-47 (claimed entering the parameters for a task of the task type into a template and executing the task type instructions associated with the at least one task on the at least one probe network device in order to form a process),. Liese also discloses that the execution server conveys protocols for successful completion of test request to custom servers that performs the requested tests, see column 4, lines 19-23. Execution Server 16 probes a test request generated by Client Machine 32 and routes that test request to the appropriate Custom Server(s) which actually performs the requested test case. See column 3, lines 29-47 and column 7, lines 9-12.

- Providing the user with the test results, see column 3, lines 48-63. (Examiner interpreted the provisioning of test results to the user as being the claimed monitoring the test network in order to determine performance).

The difference between Liese and claim 9 is that while Liese discloses conveying protocols by the execution server for successful completion of test request to custom servers that performs the requested tests, it doesn't explicitly specify translating by the customer server the parameters into instructions executable by the probe network device. (Claimed translating by the NVT server the parameters into instructions executable by the probe network device).

However, with reference to figures 4 and 5, Mann discloses a client device connected to a WEB server 406, the web server (also called host computer) configured for translating a source code user program into object code (claimed executable instructions) for execution by the target processor, the target processor hosting a test, see column 7, lines 10-61 and column 8, lines 31-44.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to implement the translation of parameters by the execution server into executable instructions to be executed by respective custom servers in the system of Liese as taught by Mann. The system of Liese can be modified by configuring the execution server to translate the parameters entered by Liese into executable instructions for transmission to a specific custom server for the respective test case for execution. The advantage would be the ability to provide the client of Liese with interface programs written in platform independent language making the clients of

Liese more versatile in testing the network. (See Mann, column 8, lines 11-14 and column 9, lines 1-10).

Regarding claim 5 and 12, Liese in view of Mann discloses substantially all the claim limitation of respective parent claims 4 and 11, further Liese discloses that all types of tests which can be performed in telecommunications are accomplished with a plethora of applications, systems, test case types, and methods, see column 2, lines 40-43. Liese in view of Mann do not specify that the call generator server (claimed traffic generator) is compatible with at least one combination of a protocol, a media and an encapsulation, wherein the protocol is selected from the group consisting of IP, IPX, CLNS, Decnet, XNS, AppleTalk, VINES, TCP, UDP, ICMP, and IGMP; the media is selected from the group consisting of Ethernet, FDDI, Serial and Token Ring; and the encapsulation is selected from the group consisting of ARPA, SNAP, SAP, Novell-Ether and HDLC.

However, Examiner take official notice that traffic generators having the feature claimed are well known in the art. It would have been obvious to a person of ordinary skill in the art to make the call generator of Liese in view of Mann being compatible with at least one combination of a protocol, a media and an encapsulation, wherein the protocol is selected from the group consisting of IP, IPX, CLNS, Decnet, XNS, AppleTalk, VINES, TCP, UDP, ICMP, and IGMP; the media is selected from the group consisting of Ethernet, FDDI, Serial and Token Ring; and the encapsulation is selected from the group consisting of ARPA, SNAP, SAP, Novell-Ether and HDLC so to provide versatility in call generators testing of Liese in view of Mann. The advantage would be

the ability to use the system/method of Liese in view of Mann to test different traffic patterns of heterogeneous networks.

Regarding claims 6, 13, Liese in view of Mann discloses substantially all the limitations of respective parent claims 4 and 11, further Liese discloses that all types of tests which can be performed in telecommunications are accomplished with a plethora of applications, systems, test case types, and methods, see column 2, lines 40-43. However Liese in view of Mann not disclose a test case being a session emulator task, wherein the session emulator task type is selected from the group consisting of a multi-protocol session emulator, a LLC2 single protocol session emulator, and a SDLC single protocol session emulator.

However Examiner take official notice that these session emulations are well known in the art.

Therefore, it would have been obvious to a person of ordinary of skill in the art at the time of the invention to enable the system/method of Liese in view of Mann to provide the known session emulations as part test cases so that versatility in session emulation in the system/method of Liese in view of Mann can be provided.

Regarding claims 7, 14, Kenner discloses substantially all the limitations of claim 4 and 11; further Liese discloses that all types of tests which can be performed in telecommunications are accomplished with a plethora of applications, systems, test case types, and methods, see column 2, lines 40-43. However Liese in view of Mann do not disclose the task type being large network emulator task type selected from the group consisting of a BGP large network emulator, a EIGRP large network emulator, an

IP RIP large network emulator, an IPX RIP large network emulator and an OSPF large network emulator.

However Examiner takes official notice that these large network emulator task types are well known in the art.

Therefore, it would have been obvious to a person of ordinary of skill in the art at the time of the invention was made to provide the system/method of Liese in view of Mann with the capability of providing these network emulations as part of the test cases so to adapt to different known network emulations techniques.

Regarding claims 8 and 15, Liese discloses substantially all the limitations of respective parent claims 4 and 11, further Liese discloses that all types of tests which can be performed in telecommunications are accomplished with a plethora of applications, systems, test case types, and methods, see column 2, lines 40-43.

However Liese in view of Mann do not explicitly disclose the task case (claimed task type) is device query task type and selected from the group consisting of a query CPU, a query memory, a query IP route, a query BGP task, a query EIGRP task, a query OSPF task, a query multi-protocol session task, a query LLC2 single-protocol session task, a query SDLC single-protocol session task, and a query traffic analyzer task.

However Examiner takes official notice that these device query task types are well known in the art.

Therefore, it would have been obvious to an ordinary person of skill in the art at the time of the invention to provide the system/method of Liese in view of Mann with the well known device query task types so that the custom servers of Liese in view of Mann

can carry out testing encompassing a variety of network devices and protocols. The advantage would be the ability to adapt the system/method of Liese in view of Mann to different network environment testing.

Regarding claim 10, Liese discloses entering the parameters for a test case among a plurality of test cases includes

- a client machine 32 (claimed NVT client) coupled to the execution server 16 (claimed coupling an NVT client to the NVT server);
- retrieving test cases by the client machine, See column 3, lines 38-39, (claimed transmitting a collection of templates corresponding to the task type to the NVT client);
- managing test cases (templates) (e.g. create, change delete, store access...), see column 3, lines 19-26 and column 3, lines 29-47, (claimed entering parameters into at least one of the collection of templates to form the task);
- communicating the test case information to the execution server, see column 3, lines 19-26. (Claimed transmitting the task to the NVT server).

Regarding claim 11, Liese discloses that a custom server is CG server (call generator), see column 3, lines 17-19, column 6, lines 34-35.

Regarding claim 16, Liese doesn't disclose that the client and custom server coupled through the Internet and the templates and the task are transmitted using JAVA/HTML.

However, with reference to figure 4, Mann discloses an Internet connection 404 between the remote terminal (client) and the computer server, wherein the programs to

be tested are transmitted using JAVA, see column 8, lines 11-14, and column 9, lines 1-9.

Therefore, it would have been obvious to a person of ordinary of skill in the art at the time the invention was made to implement the JAVA protocol along the Internet connection as taught by Mann in lieu of the client/server architecture of Liese so that Liese testing apparatus/method be adapted to an Internet environment. The advantage would be using the known JAVA browsing in carrying out the testing of Liese by downloading the task cases (templates) and carrying remote testing over the Internet.

Regarding claim 17, with reference to figures 1 and 2, Liese discloses: user at the network under test communicates to a client machine which test or test cases are to be executed by custom servers, the test or test cases can be edited before transmission to the execution server 16 which coordinates the execution of test cases by the custom servers, see column 3, lines 9-47. (Claimed forming a task, the task being formed by entering task parameters into a task template). Liese further discloses that the client machine 32 includes a GUI (Graphical User Interface) that provides an interface for managing test cases to be executed by the custom servers (e.g. create, change delete, store access...), see column 3, lines 29-47. Wherein an Execution Server 16 probes a test request generated by Client Machine 32 and routes that test request to the appropriate Custom Server(s) which actually performs the requested test case. See column 7, lines 9-12. Liese also discloses that the execution server conveys protocols for successful completion of test request to custom servers that performs the requested tests.

The difference between Liese and claim 17 is that Liese doesn't explicitly specify translating the task parameters using the execution server to form executable instructions by a custom server, the customer server host a task code for executing the executable instruction.

However, with reference to figures 4 and 5, Mann discloses a client device connected to a WEB server 406, the web server (also called host computer) configured for translating a source code user program into executable instructions for execution by the target processor, see column 7, lines 10-61 and column 8, lines 31-44. (Examiner interpreted the capability of the target processor for executing the executable instructions to read on the claimed the probe network device hosting a task code for executing the executable instructions).

Therefore, it would have been obvious to a person of ordinary of skill in the art at the time the invention was made to implement the translation of parameters by the execution server into executable instructions to be executed by respective custom servers in the system of Liese as taught by Mann. The system of Liese can be modified by configuring the execution server to translate the parameters entered by Liese into executable instructions for transmission to a specific custom server for the respective test case for execution. The advantage would be the ability to provide the client of Liese with interface programs written in platform independent language making the clients of Liese more versatile in testing the network. (See Mann, column 8, lines 11-14 and column 9, lines 1-10).

Regarding claim 18 and 20, Liese discloses that one of the test cases transmitted to one of custom servers, the customer server being CG server (call generator), (see Liese, column 3, lines 17-19, column 6, lines 34-35, but doesn't disclose the test case(s) is/are selected from a group of tasks consisting of a traffic generator, a traffic analyzer, a large network emulator, a session emulator, a device query or a script task.

However, Examiner takes official notice that these tasks are well known in the art.

It would have been obvious to a person of ordinary skill in the art to make the task of Liese in view of Mann being selected from a group of a traffic generator, a traffic analyzer, a large network emulator, a session emulator, a device query or a script task so that the testing system/method of Liese in view of Mann can be used in a variety of network testing.

Regarding claim 19, with reference to figures 1 and 2, Liese discloses:

a user at the network under test communicates to a client machine which test or test cases are to be executed (claimed sending task templates to a user) by the custom servers, the test or test cases can be edited before transmission to the execution server 16 (claimed receiving tasks formed by the user entering parameters into the task templates) which coordinates the execution of test cases by the custom servers, see column 3, lines 9-47. Liese also discloses that the execution server conveys protocols for successful completion of test request to custom servers that performs the requested tests. See column 3, lines 29-47, and column 4, lines 19-23.

The difference between Liese and claim 20 is that Liese doesn't explicitly disclose translating the task to a task code configured to be executed by one or more custom servers. (Claimed translating the tasks to task code configured to be executed by one ore more probe network devices).

However, with reference to figures 4 and 5, Mann discloses a client device connected to a WEB server 406, the web server (also called host computer) configured for translating a source code user program into executable instructions for execution by the target processor, see column 7, lines 10-61 and column 8, lines 31-44.

Therefore, it would have been obvious to a person of ordinary of skill in the art at the time the invention was made to implement the translation of parameters by the execution server into executable instructions to be executed by respective custom servers in the system of Liese as taught by Mann. The system of Liese can be modified by configuring the execution server to translate the parameters entered by Liese into executable instructions for transmission to a specific custom server for the respective test case for execution. The advantage would be the ability to provide the client of Liese with interface programs written in platform independent language making the clients of Liese more versatile in testing the network. (See Mann, column 8, lines 11-14 and column 9, lines 1-10).

Regarding claim 21, as indicated above with reference to parent claim 2, Liese discloses client 32 includes a GUI (Graphical User Interface) that provides an interface for managing test cases (e.g. create, change delete, store access...), and supplying the edited test case to the execution server, see column 8, lines 14-18. Liese further

discloses the user retrieves tests cases to be edited (claimed collection of templates) from different locations such as local memory or shared storage, see column 3, lines 48-50, column 7, lines 49-60.

The difference between Liese and claim 21 is that Liese doesn't specify retrieving the test cases (collection of templates) by the execution server. (Claimed the NVT server is configured to transmit a collection of templates to the NVT client).

However, Mann discloses downloading from the host computer (claimed NVT server as discussed in the base claim 1 above) an interface program on a local computer for setting test parameters, see figure 5, steps 512-516, and column 9, lines 4-9.

Therefore it would have been obvious to a person of ordinary skill in the art, at the time the invention was made to modify the system of Liese of templates collection provisioning by the execution server in stead of the shared database as indicated by Mann so to localize both the management of the customer servers and shared database. The advantage would be the ability to locally monitor and trouble-shoot the execution server of Liese in view of Mann.

Response to Arguments

2. Applicant's arguments filed 01/17/2007 have been fully considered but they are not persuasive:

As to claim 1:

Applicants alleged that neither Liese nor Mann provides disclosure of the claimed limitations "configured to translate parameters" and "configured to transmit the instructions".

Applicants stated that *Mann's IDE purportedly provides for loading a program from a remote terminal to a host computer that then translates the source code of the user program into object code for execution by a target processor*. Applicants further alleged that *Mann's target processor is disclosed to be a microcontroller or microprocessor for which the remote user is testing their program code. Mann provides no indication that such a processor corresponds to a "probe network device coupled to the network under test," as claimed. In fact, there is no indication at all that the disclosed target processor is coupled to any network. Further, there is no indication within Mann that the target processor is "configured to host at least one task type."*, Applicants submit that Mann cannot be said to provide disclosure of *"the NVT server is configured to transmit the instructions to the at least one probe network device hosting the task type," as claimed. Emphasis added.*

Applicants alleged that *Mann provides no disclosure that the target processor is "configured to execute a process corresponding to the at least one task type in response to the instructions."* Applicants concluded that *Mann only provides that the source code provided by the user is translated into object code that is ultimately executed by the target processor. (Mann 8:38-40). Thus, Mann teaches away from the target processor executing anything else but the translated source code in response to receiving such code. Emphasis added.*

Examiner respectfully disagrees with Applicants' position. Examiner already stated in the office action that Liese doesn't disclose the limitations of "configured to translate parameters" and "configured to transmit the instructions". As to Mann and contrary to Applicants, Mann doesn't need to teach: *"probe network device coupled to the network under test," the target processor is "configured to host at least one task type." and "the NVT server is configured to transmit the instructions to the at least one probe network device hosting the task type,"*. Because, Liese already teaches such features. What is relevant about Mann, is that the host computer *translates the source code of the user program (from a client computer) (claimed template with entered parameters) into object code for execution by a target processor (claimed transmit the executable instructions"*. The system of Liese can be modified by configuring the execution server to translate the parameters entered by Liese into executable instructions for transmission to a specific custom server for the respective test case for execution as taught by Mann. The advantage would be the ability to provide the client of Liese with interface programs written in **platform independent language**, making the clients of Liese more versatile in tasking the network under test. (See Mann, column 8, lines 11-14 and column 9, lines 1-10). Therefore, and contrary to Applicants assumption of non-obviousness of Liese in view of Mann, the test of obviousness is what the combined teachings of the references would have suggested to those of ordinary skill in the art. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981).

As to the argument of impermissible hindsight on page 11 of the remarks, and in response to Applicants' argument that the examiner's conclusion of obviousness is

based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

Applicants on last paragraph, page 11, further argued that *Liese already discloses that the disclosed client machines are platform independent, therefore the Office Action suggestion that such a combination would "provide the client of Liese with interface programs written in platform independent language making the clients of Liese more versatile in testing the network."* doesn't stand of being proper suggestion. Applicant further stated that *Liese provides no disclosure of the desirability of passing executable instructions from the execution server to the various disclosed custom servers. Liese's execution server is disclosed to merely prioritize jobs and then queue them to the custom servers. The parameters provided by the client machines are disclosed to be usable by the custom servers. Indeed, the execution server is disclosed to access little to none of the parameters actually provided by the client machines, because such information is disclosed to be stored in various databases or other shared storage. (Liese 7:32-8:11).*

In response, Examiner notes that Applicants have mischaracterized the teaching of Liese in combination with Mann. Liese with reference to figure 1, discloses

A Client/Server Test Architecture may be used in either a LAN (local area network) or WAN (wide area network) environment, or any other computer environment. Bus 12 can be configured in any topology desired, such as a ring or daisy chain topology, independent of the LAN or WAN topology of the system so long as there is a communications link between Client Machine(s) 18 and Execution Server 16 and a communications link between Execution Server 16 and Custom Server(s) 14. See Liese column 5, lines 17-31. Liese doesn't specify communications link between Client Machine(s) 18 and Execution Server 16 and a communications link between Execution Server 16 and Custom Server(s) 14 are **Internet** links. The mischaracterization of Applicants to the combined references is that Mann actually discloses (in the addition to the limitations argued above: host computer *translating the source code of the user program into object code for execution by a target processor*) an interface programs written in platform independent language that is java language (as an example), which is not the same as platform independency of Liese as Applicants alleged, the client machines of Liese do not use platform independent language (such as java) as contended by Applicants . Therefore, Liese lack of specifying performing the tests using a java compliant interface is a clear and strong motivation for a person of skill in the art to use Mann's remote testing over the Internet from any machine regardless of associated platform **language**. The success of such combination is evident by recognizing the easy implementation of the translation feature of codes into executable tasks by incorporating such translation into the

execution server of Liese as taught by Mann as another layer along the coordination function of the execution server of Liese.

Applicants further alleged that "*Liese provides no disclosure of the desirability of passing executable instructions from the execution server to the various disclosed custom servers*". Applicants further argue that the Office Action fails to *provide any argument supporting a likelihood of success in combining Liese with Mann. Liese provides disclosure of a system that can purportedly receive test case requests from the disclosed client machines, prioritize and queue those test case requests using the disclosed execution server, and then providing those test cases per their priority to the disclosed custom servers. There is no disclosure within Liese that the information provided by the client machines is in the form of "source code" such as that described by Mann. Nor is there any disclosure within Liese of the need to transform the test case requests received from the various client machines into executable code for the custom servers, as suggested by the compiling that occurs within the IDE of Mann. In addition, there is no indication within Mann that the disclosed target processors can be used to execute network test cases such as those described by Liese. Nor is there any indication within Mann that the IDE can merely prioritize and queue untranslated task requests such as those provided by Liese's client machines*".

Examiner respectfully disagrees. In response to Applicants' arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091,

231 USPQ 375 (Fed. Cir. 1986). More specifically, Liese alone *doesn't* need to provide the desirability of passing executable instructions from the execution server to the various disclosed custom servers, and/or the information provided by the client machines is in the form of "source code", and the need to transform the test case requests received from the various client machines into executable code for the custom servers; in the other hand Mann alone doesn't need to provide for executing network test cases such as those described by Liese, because Liese does provide for executing network test cases.

However, Mann discloses a) passing executable instructions from the execution server to the various target processors, b) the information provided by the client machines is in the form of "source code", and c) transforming the test case requests received from the various client machines into executable code for target processors.

It is the combination of the teaching of Mann with that of Liese that render the claim unpatentable. It should be noted for the sake of the argument that given the most reasonable interpretation of the claim limitations. Mann alone can be regarded as anticipating the invention as claimed (claim 1), because the client machine of Mann provides for entering parameters into templates and forwarding the template to the web server (host computer) where the translation take place of the source codes into executable codes to be executed by a selected target processor. The reasons behind this anticipation may be drawn from the fact that limitations appearing in the specification but not recited in the claim should not be read into the claim. *E-Pass Techs., Inc. v. 3Com Corp.*, 343 F.3d 1364, 1369, 67 USPQ2d 1947, 1950 (Fed. Cir.

2003) (claims must be interpreted "in view of the specification" without importing limitations from the specification into the claims unnecessarily). In re Prater, 415 F.2d 1393, 1404-05, 162 USPQ 541, 550- 551 (CCPA 1969). See also In re Zletz, 893 F.2d 319, 321-22, 13 USPQ2d 1320,1322 (Fed. Cir. 1989).

Nevertheless, as noted above, the teaching of Mann and its advantage of remote testing over the Internet and its platform independent language would motivate an ordinary person of skill in the art, at the time the invention was made to modify Liese to incorporate the feature of Mann to come up with the invention as claimed.

Claims 2-8 and 21:

Applicants argue that claims 2-8 and 21 depend from claim 1, and thus they are patentable. Examiner respectfully disagrees for the reason given above with regard to argument and rejections above.

Claims 9,17 and 19:

Applicants traverse the rejections of claims 9, 17 and 19 over Liese in view of Mann, by incorporating the discussion with regard to claim 1. In response Examiner traverse such argument for similar reasons indicated above with regard to claim 1.

Claims 6 and 13:

Applicants traversed Examiner official notice of indicating that there is no suggestion made that Liese or Mann disclose a session emulation task type. Examiner disagrees for the following reasons. Liese alone discloses that **all types of tests which can be performed in telecommunications are accomplished with a plethora of applications, systems, test case types, and methods, see column 2, lines 40-43.**

Therefore, it is within the teaching of Liese to have the test case type being as an example of session emulator task, wherein the session emulator task type is selected from the group consisting of a multi-protocol session emulator, a LLC2 single protocol session emulator, and a SDLC single protocol session emulator, as evidenced by Liese's **all types of tests** that are accomplished with a **plethora test case types**.

Therefore, Examiner respectfully concludes that Applicants specific emulator task types can be clearly described as only one of non-limiting examples of the test case types of Liese.

Claims 7, 8, 14 and 15:

Applicants traversed Examiner rejection of these claims asserting that neither Liese nor Mann, nor the Official Notice taken in the Office Action, provides disclosure of all of the limitations of Claims 7, 8, 14 and 15 and all claims depending therefrom. In particular Applicants stated that there is no large network emulator (claims 7 and 14), and device query tasks (claims 7 and 14). Examiner respectfully disagrees, for similar reasons indicated above with regard to claims 6 and 13.

Conclusion

3. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not

Art Unit: 2616

mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to AHMED ELALLAM whose telephone number is (571) 272-3097. The examiner can normally be reached on 9-5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Seema Rao can be reached on (571) 272-3174. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

A.E.
Examiner
Art Unit 2616
4/6/07

Seema S. Rao
SEEMA S. RAO 4/12/07
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600